Amendment to the Claims

The listing of claims below will replace all prior versions and listings of claims in the application.

1. (Previously Presented) A background memory manager (BMM) for managing a memory in a data processing system, the BMM comprising:

circuitry to transfer a data structure to and from an outside device and to and from a memory; and

a communication link;

management logic coupled to a processor by the communication link and configured to determine if the data structure fits into the memory, to decide where, in a plurality of regions in the memory, to store the data structure, to perform data transfers between the outside device and the memory, to maintain a memory state map according to memory transactions made, and to inform the processor of new data and its location.

2. (Previously Presented) The BMM of claim 1, wherein the management logic is further configured to provide, for the data structure and when the data structure is stored, a data identifier on the communication link.

3. (Previously Presented) The BMM of claim 2, wherein the management logic is further configured to update, in response to the memory transactions, the memory state map to a new memory state to keep track of regions occupied by valid data and regions unoccupied by valid data.

4. (Currently Amended) The BMM of claim 2, wherein the management logic is further configured to copy, in response to a signal on the communication link that the processor is finished with identified data in the memory, the identified data from the memory to another device and to update the memory state map to indicate a new memory state for a region of the identified ~~data~~ data.

5. (Previously Presented) The BMM of claim 1, further comprising an interrupt handler configured to allow a remote data source to interrupt the BMM when data is available to be transferred to the memory.

6. (Previously Presented) The BMM of claim 1, wherein data handled by the BMM constitutes network data packets.

7. (Previously Presented) A data processing system, comprising:

a processor;

a memory coupled to the processor; and

a background memory manager (BMM) coupled to the memory and the processor, the BMM including circuitry to transfer a data structure to and from an outside device and to and from the memory and including management logic configured to determine if the data structure fits into the memory, to decide where, in a plurality of regions in the memory, to store the data structure, to perform data transfers between the outside device and the memory, to maintain a memory state map according to memory transactions made, and to inform the processor of new data and its location.

8. (Previously Presented) The data processing system of claim 7, wherein the management logic is further configured to provide, for the data structure and when the data structure is stored, a data identifier to the processor.

9. (Previously Presented) The data processing system of claim 8, wherein the management logic is further configured to update, in response to the memory transactions, the memory state map to a new memory state to keep track of regions occupied by valid data and regions unoccupied by valid data.

10. (Currently Amended) The data processing system of claim 8, wherein the management logic is further configured to copy, in response to a signal from the processor that the processor is finished with identified data in the memory, the identified data from the memory to another device and to update the memory state map to indicate a new memory state for a region of the identified ~~data~~ data.

11. (Previously Presented) The data processing system of claim 7, further comprising an interrupt handler configured to allow a remote data source to interrupt the BMM when data is available to be transferred to the memory.

12. (Previously Presented) The data processing system of claim 7, wherein data handled by the BMM constitutes network data packets.

13. (Previously Presented) A network packet router, comprising:

an input/output (I/O) device configured to receive and to send a packet on the network;

a processor;

a memory coupled to the processor; and

a background memory manager (BMM) coupled to the memory and the processor, the BMM including circuitry configured to transfer the packet to and from the I/O device and to and from the memory and including management logic configured to determine if the packet fits into the memory, to decide where, in a plurality of regions in the memory, to store the packet, to perform data transfers between the I/O device and the memory, to maintain a memory state map according to memory transactions made, and to inform the processor of new data and its location.

14. (Previously Presented) The data router of claim 13, wherein the management logic is further configured to provide, for the packet and when the packet is stored, a data identifier to the processor.

15. (Previously Presented) The data router of claim 14, wherein the management logic is further configured to update, in response to the memory transactions, the memory state map to a new memory state to keep track of regions occupied by valid packets and regions unoccupied by valid packets.

16. (Currently Amended) The data router of claim 14, wherein the management logic is further configured to copy, in response to a signal that the processor is finished with a first

packet in the memory, the first packet from the memory to the I/O device and to update the memory state map to indicate a new memory state for a region of the first ~~packet~~ packet.

17. (Previously Presented) The data router of claim 13, further comprising an interrupt handler configured to allow the I/O device to interrupt the BMM when packets are available to be transferred to the memory.

18. (Previously Presented) A method for managing a memory in a data processing system having a processor, the method comprising:

(a) transferring a data structure to and from an outside device and to and from the memory by circuitry in a background memory manager (BMM);

(b) determining by the BMM if the data structure from the outside device will fit into available space in the memory;

(c) deciding by the BMM exactly where, in a plurality of regions in the memory, to store the data structure; and

(d) updating a memory state map in the BMM when a memory transaction is made.

19. (Previously Presented) The method of claim 18, further comprising providing, for the data structure and when the data structure is stored, a data identifier on a link to the processor.

20. (Previously Presented) The method of claim 19, further comprising keeping track of regions occupied by valid data and regions unoccupied by valid data.

21. (Previously Presented) The method of claim 19, further comprising:

copying, in response to a signal that the processor is finished with identified data in the memory, the identified data from the memory to another device; and

updating the memory state map to indicate a new memory state for a region of the identified data.

22. (Previously Presented) The method of claim 18, further comprising interrupting the BMM by the outside device when data is available to be transferred to the memory.

23. (Previously Presented) The method of claim 18, wherein data handled by the BMM constitutes network data packets.

24. (Previously Presented) The method of claim 23, wherein the network data packets are conveyed on the Internet.

25. (Previously Presented) The data router of claim 13, wherein the BMM is further characterized by being configured to notify the processor when enough of the packet is stored for the processor to begin to perform desired processing.